LISTING OF THE CLAIMS

Claims 1-18 **(canceled)**

19. **(currently amended)** A method, implemented in a computer system, of shifting a multi-word value comprising:

performing a first shift operation on a first portion of the multi-word value to produce one or more overflow bits;

performing a second shift operation on a second portion of the multi-word value, where the second shift operation comprises:

producing a shift result; and

concatenating the shift result and the overflow bits; and

where the second shift operation is a multi-precision shift instruction and where the first shift ~~instruction~~ operation and second shift ~~instruction~~ operation are performed sequentially and further where the one or more overflow bits are stored in an overflow register prior to concatenation with the shift result.

20.**(currently amended)** The method of claim 19 [[,]] ~~where the second shift operation is a multi-precision shift instruction, and where the second shift operation produces a result, the method~~ further comprising:

fetching and decoding the multi-precision shift instruction; and

outputting the result.

21. **(previously presented)** The method of claim 20, where the multi-precision shift instruction is a shift left instruction.

22. **(previously presented)** The method of claim 20, where the multi-precision shift instruction

is a shift right instruction.

23. **(previously presented)** The method of claim 20, where the multi-precision shift instruction specifies a shift increment.

24. **(previously presented)** The method of claim 23, where the shift increment is greater than or equal to the number of bits in a word.

25. **(previously presented)** The method according to claim 23, where the shift increment is less than the number of bits in a word.

26. **(previously presented)** The method of claim 19, further comprising:

storing one or more bits shifted out of the second portion of the multi-word value during the second shift instruction in a carry register.

27. **(previously presented)** The method of claim 19, where concatenating the shift result and the overflow bits comprises:

performing a logical OR operation on at least one bit in the shift result and at least one overflow bit.

28. **(previously presented)** The method of claim 19, further comprising:

storing one or more of the overflow bits in a carry register.

29. **(currently amended)** A processor for processing multi-precision shift instructions, comprising:

a program memory for storing instructions including at least one multi-precision shift instruction;

a program counter for identifying current instructions for processing; and

a barrel shifter for executing shift instructions, including the at least one multi-precision shift instruction, the barrel shifter including:

one or more carry registers for storing values shifted out of sections of the barrel shifter; and

OR logic for concatenating values stored in one or more carry registers with values in the barrel shifter; and

where the barrel shifter is operable to shift a multi-word value, and where when shifting a multi-word value the barrel shifter:

executes at least one shift instruction to:

load a first operand into a section within the barrel shifter, where the first operand is a first portion of the multi-word value; and

generate one or more overflow bits;

storing the one or more overflow bits in an overflow register and

executes at least one multi-precision shift instruction fetched from the program memory to:

load a second operand into a section within the barrel shifter, where the second operand is a second portion of the multi-word value;

shift the second operand;

concatenate the second operand with one or more of the overflow bits; and

output the shifted value.

30. **(previously presented)** The processor of claim 29, where the multi-precision shift instruction is a shift left instruction.

31. **(previously presented)** The processor of claim 29, where the multi-precision shift instruction is a shift right instruction.

32. **(previously presented)** The processor of claim 29, where the multi-precision shift instruction is an arithmetic shift instruction.

33. **(previously presented)** The processor of claim 29, where the multi-precision shift instruction is a logical shift instruction.

34. **(previously presented)** The processor of claim 29, where the multi-precision shift instruction specifies a shift increment.